

What is claimed is:

1. An incremental remote loading apparatus, comprising:

5 a dependent reader module for receiving a cross-compiled object file from a program development tool, analyzing the cross-compiled object file according to a type of an object file and detecting independent linking information from the type of the object file; and

10 an independent linker module for receiving the detected linking information from said dependent reader module, downloading the object file to a target system by using the detected linking information and rearranging target modules of the target system.

15 2. The incremental remote loading apparatus as recited in claim 1, wherein said dependent reader module includes:

20 a COFF reader module for receiving a cross-compiled COFF object file from said program development tool, analyzing the object file dependently to a COFF type of the object file and detecting independent linking information from the type of the object file; and

25 an ELF reader module for receiving a cross-compiled ELF object file from the program development tool, analyzing the object file dependently to an ELF type of the object file and detecting independent linking information from the type of the object file.

3. The incremental remote loading apparatus as recited in claim 1, wherein the linking information includes section information, symbol information and rearrangement information.

5 4. The incremental remote loading apparatus as recited in claims 3, wherein said linker module rearranges not only object files being loaded but also loaded target modules and provides an incremental remote linking, which links object files to the target system without a linking order of the  
10 object files.

5. An incremental remote loading method, comprising the steps of:

a) at a reader module, analyzing necessary linking  
15 information for linking object files;

b) at a linker, allocating a target memory space for sections according to a section information;

c) determining whether each entry of a symbol table is defined or not and calculating addresses of sections in a  
20 target memory;

d) determining, according to a result of said step c), whether a symbol defined or not in case said symbol is stored in said symbol table or inserting a new symbol to the symbol table in case the symbol is not in said symbol table and  
25 determining whether the new symbol is defined or not;

e) rearranging an object file if a symbol is defined or rearranging the object file after transforming a defined

symbol in case the symbol is not defined; and

f) transmitting a rearranged object file to a target memory.

5           6. The incremental remote loading method as recited in claim 5, wherein said step a) includes the steps of:

a1) determining a type of the cross-compiled object file;

a2) analyzing the linking information by connecting a COFF reader in case said object file type is a COFF type; and

10          a3) analyzing the linking information by connecting an ELF reader in case said object file type is an ELF type.

7. The incremental remote loading method as recited in claim 5, wherein said step d) includes the steps of:

15          d1) generating a new symbol if an entry is not stored in said symbol table and is not a defined symbol;

d2) adding symbol information including a symbol name and an address of a target memory to the new generated symbol; and

20          d3) registering and inserting the symbol, which is generated and the symbol information added, to the symbol table.

8. The incremental remote loading method as recited claim 5, wherein transforming undefined symbol to defined symbol process in said step e) includes the steps of:

e1) transforming an undefined symbol to an defined symbol

if a symbol in a symbol table is undefined; and

e2) adding an address of a target memory and rearranging modules of a target system by using a rearrangement information of said undefined symbol.

5

9. The incremental remote loading method as recited in claim 5, wherein said step d) includes the steps of:

d1) bringing a rearranged symbol to said symbol table;

d2) rearranging a text and data sections in a host  
10 system base on a rearrangement information of an entry and an address of a target memory of a symbol in case said brought symbol is defined; and

d3) adding a rearrangement information to said symbol incase said brought symbol is undefined.

15

10. A computer-readable recording medium storing instruction for executing an incremental remote loading method, comprising the functions of:

a) analyzing necessary linking information for liking  
20 object files by a reader modules;

b) allocating a target memory space for sections according to section information by a linker;

c) determining whether each entry of a symbol table is defined or not and calculating addresses of sections in a  
25 target memory;

d) determining, according to a result of said step c), whether a symbol defined or not in case said symbol is stored

in said symbol table or inserting a new symbol to the symbol table in case the symbol is not in said symbol table and determining whether the new symbol is defined or not;

e) rearranging an object file if a symbol is defined or  
5 rearranging the object file after transforming a defined symbol in case a symbol is not defined; and

f) transmitting a rearranged object file to a target memory.

10